

# Mikhail Raskin

Curriculum vitae

## Personal details

<b>Date and place of birth</b>	21 August 1987, Moscow, Russia
<b>Nationality</b>	Russia
<b>Address</b>	University of Bordeaux, LaBRI (Bât A30), 351, Cours de la Libération, F-33405 Talence Cedex, France
<b>Phone</b>	+33 7 69 55 11 75
<b>Email</b>	mraskin@labri.fr, raskin@mccme.ru

## Current position

Since 2017 Research Engineer, University of Bordeaux, LaBRI

## Research interests

Theoretical computer science; expressive possibilities for Human–Computer and Program–Program interaction; data analysis; algorithms.

## Education

**2008–2014** PhD, Moscow State University, Dept. of mechanics and mathematics

**PhD thesis supervisor:** Prof. Nikolay K. Vereshchagin

**PhD thesis title:** Automata on infinite words: direct and semidirect products approach.

**2002–2008** Master of Science, Independent University of Moscow

**2003–2008** Master of Science, Moscow State University, Dept. of mechanics and mathematics (cum laude)

**MSc thesis supervisor:** Prof. Nikolay K. Vereshchagin

**MSc thesis title:** Partial orderings on measures on the set of infinite words over a two-symbol alphabet

**1999–2003** Moscow State School N 57

## Publications

1. Ivan Damgård, Jesper Buus Nielsen, Antigoni Polychroniadou, Michael Raskin. On the Communication Required for Unconditionally Secure Multiplication. *Advances in Cryptology - CRYPTO Proceedings* 2016, part 2, pp. 459–488.

We present the following results:

In the honest majority setting, as well as for dishonest majority with preprocessing, any gate-by-gate protocol must communicate  $\Omega(n)$  bits for every multiplication gate, where  $n$  is the number of players.

In the honest majority setting, we show that one cannot obtain a bound that also grows with the field size. Moreover, for a constant number of players, amortizing over several multiplication gates does not allow us to save on the computational work, and – in a restricted setting – we show that this also holds for communication.

My contribution is the method of proving the communication lower bound for amortizing over several computations and the method for ruling out a good lower bound for multiplication over a large field.

2. K. Chatterjee, M. Henzinger, S. Krinninger, V. Loitzenbauer, M. Raskin. Approximating the minimum cycle mean. *Theoretical Computer Science*, 2014, vol. 547, pp. 104–116

A fast method for finding a cycle of approximately minimal mean edge weight in a directed graph with non-negative edge weights is offered using previously known approximate matrix multiplication methods.

My contribution is the method of finding an example of such a cycle in addition to finding the approximate length without increasing asymptotical upper bound on the run time.

3. M. Raskin. Toom's Partial Order Is Transitive. *Problems of Information Transmission*, 2012, vol. 48, N2, pp.154–172

The main result of the article is the proof of transitivity of Toom's order relation on measures on bi-infinite sequences of pluses and minuses. The relation was initially proposed as a tool to study ergodicity of cellular automata with deletions.

4. M. Raskin. Coupling of computable measures coordinated with an order relation is not always computable. *Vestnik Moskovskogo Universiteta (MSU Bulletin)*. Series 1. Mathematics. Mechanics. 2012, v. 2

An example is given of two computable measures on infinite binary sequences such that there is a measure on pairs of infinite sequences such that the first projection of a random pair is almost surely larger than the second one and the projections of the measure are equal to original measures, but all such measures on pairs are uncomputable.

5. M. Raskin. Lower Estimate of the Regulator of the Direct Product of almost Periodic and Periodic Sequences. *Vestnik Moskovskogo Universiteta (MSU Bulletin)*, Series 1, Mathematics. Mechanics. 2011, v. 6

A lower bound is proved for almost periodicity regulator of almost periodic sequence coupled with a periodic sequence. This lower bound differs from known upper bound only in a multiplicative constant of in the count of iterations of regulator of almost periodicity of original sequence.

## Conference talks

### Reviewed talks

1. Writing a best-effort portable code walker in Common Lisp. European Lisp Symposium 2017, Brussels, Belgium.

One of the powerful features of the Lisp language family is possibility to extend the language using macros. Some of possible extensions would benefit from a code walker, i.e. a library for processing code that keeps track of the status of different part of code, for their implementation. But in practice code walking is generally avoided.

In this paper, we study facilities useful to code walkers provided by “Common Lisp: the Language” (2nd edition) and the Common Lisp standard. We will show that the features described in the standard are not sufficient to write a fully portable code walker.

One of the problems is related to a powerful but rarely discussed feature. The `macrolet` special form allows a macro function to pass information easily to other macro invocations inside the lexical scope of the expansion.

Another problem for code analysis is related to the usage of non-standard special forms in expansions of standard macros. We review the handling of `defun` by popular free software Common Lisp implementations.

We also survey the abilities and limitations of the available code walking and recursive macro expansion libraries. Some examples of apparently-conforming code that exhibit avoidable limitations of the portable code walking tools are provided.

We present a new attempt to implement a portable best-effort code walker for Common Lisp called Agnostic Lizard.

2. Paradox of choice in social network games with product choice. (joint work with Nikita Nikitenkov) Game theory society congress 2016, Maastrich, Netherlands.

The definition of product choice games on social networks was introduced by K. Apt and S. Simon. In an article written jointly with E. Markakis, they considered four types of paradox of choice in such games and gave examples of three of them. The existence of paradoxical networks of the fourth type was proven only in a weakened form. The existence of so-called «vulnerable networks» in the strong sense remained an open question.

In the present paper we solve this open question by introducing a construction, called a «cascade», and use it to provide uniform examples for all four definitions of paradoxical networks.

3. Accessing local variables during debugging. (joint work with Nikita Mamardashvili) European Lisp Symposium 2016, Krakow, Poland.

Any reasonably large program has to use local variables. It is quite common in the Lisp language family to also allow functions that exist only in a local scope. Scoping rules often allow compilers to optimize away parts of the local environment; doing that is good for performance, but sometimes inconvenient for debugging.

We present a debugging library for Common Lisp that ensures access to the local variables during debugging. To prevent the optimisations from removing access to these variables, we use code-walking macros to store references to the local variables (and functions) inside global variables.

4. Computable measures that are couplable but not computably couplable. International Conference on Computability, Complexity and Randomness 2013, Moscow, Russia.
5. Data-transformer: an example of data-centered tool set. European Lisp Symposium 2013, Madrid, Spain.

This paper describes the data-transformer library, which provides various input and output routines for data based on a unified schema. Currently, the areas of the library's use include storage and retrieval of data via CLSQL; processing CSV and similar tabular files; interaction with user via web forms. Using the supplied schema, the data-transformer library can validate the input, process it and prepare it for output. A data schema may also include channel-specific details, e.g. one may specify a default HTML textarea size to use when generating the forms.

6. Almost periodicity and finite automata (joint work with Yu. Pritykin). Workshop on Infinite Words, Automata and Dynamics 2007, Ekaterinburg, Russia.

### **Lightning talks**

1. Lisp-in-the-middle, or I wanted a Lisp Machine and all I got was a fancy sudo. Lightning talk. European Lisp Symposium 2017, Brussels, Belgium.

This talk described issues making Lisp Machine approach unfeasible for a modern workstation, and presents an approach to using Lisp code to describe policies while using software in other languages as black box components and isolating components from each other.

2. OpenCV wrapper and form processing for Common Lisp — implementation notes. Lightning talk. European Lisp Symposium 2015, London, UK.

This talk contained implementation notes for the tools related to processing scanned submissions at some events. For example, some of these tools were developed to recover QR-codes incorrectly processed during scanning by the scanning software.

3. Julia: an outside view. Lightning talk. European Lisp Symposium 2014, Paris, France.

A small overview of the Julia programming language by a Common Lisp and Julia user for other Lisp users.

4. Screen subareas in StumpWM via tagging. Lightning talk. European Lisp Symposium 2014, Paris, France.

This talk presented a set of StumpWM modules to change window management logic using tagging instead of putting each window in a group.

5. Personal software for PC: how I ended up reading email using Common Lisp. Lightning talk. European Common Lisp Meeting 2013, Madrid, Spain.
6. QueryFS: a virtual filesystem based on queries and related tools. Lightning talk. European Lisp Symposium 2012, Zadar, Croatia.

These two talks described QueryFS.

QueryFS is a virtual filesystem for POSIX-like systems (its development platform is SBCL on Linux) that represents results of user-supplied queries as directories. My main current use-case is using it to navigate and process data stored or indexed in PostgreSQL with traditional tools (grep, find, vim etc.)

## Conferences attended

1. European Lisp Symposium 2017, Brussels, Belgium
2. Game theory society congress (GAMES) 2016, Maastricht, Netherlands
3. Logical foundations of game theory (LOFT) 2016, Maastricht, Netherlands
4. European Lisp Symposium 2016, Krakow, Poland
5. European Lisp Symposium 2015, London, UK
6. Games, Automata, Logic and Formal Verification (GandALF) 2014, Verona, Italy
7. European Lisp Symposium 2014, Paris, France
8. Computability, Complexity and Randomness 2013, Moscow, Russia
9. Games, Automata, Logic and Formal Verification (GandALF) 2013, Borca di Cadore, Italy
10. European Lisp Symposium 2013, Madrid, Spain
11. European Lisp Symposium 2012, Zadar, Croatia
12. Computer Science in Russia 2007, Ekaterinburg, Russia

## Scholarships

**2005–2008** Kolmogorov Scholarship for students in mathematical logic and theory of algorithms

## Computer experience

**Main languages:** Common Lisp, Python, Pascal (Free Pascal Compiler, Delphi),  
POSIX Shell/Bash, Nix

**Other:** Julia, Scheme, C, JavaScript, OCaml, C++

**System administration:** GNU/Linux servers; small networks with complicated routing

## Professional programming

- 2008–2015** Design and implementation of web-based software systems for data collection, automated validation, unification and analysis. (MCCME)
- 2004–2008** Development of software systems for automated data acquisition, processing and monitoring in engineering tests. (NATI)

## Contributions to open source projects

<b>NixPkgs</b>	import software packages to the NIX platform, minor patches to the Nix package manager
<b>Agnostic Lizard</b>	author and current maintainer of a code walking library for Common Lisp
<b>StumpWM</b>	tag-based window management code for Stump Window Manager
<b>CL-Emb</b>	current maintainer
<b>RelFS</b>	symlink generation code for Relational filesystem
<b>FunionFS</b>	support for dynamical reconfiguration of the union filesystem

## Teaching experience

Independent University of Moscow (2007–2015):

Lecture courses: Introduction to probability theory, Set theory, Mathematical logic

Teaching assistant: Geometry, Algebra, Mathematical analysis

Moscow Institute of Physics and Technology (2011–2014):

Teaching Assistant: Mathematical logic, Algorithmic complexity.

Lecture Courses at the Summer School “Contemporary Mathematics” (2007–2015)

- “We cannot wait for favors from Nature” (forcing method in set theory model construction);
- “Who am I? Where am I?” (on sampling assumptions);
- A set of conventions and conventions about sets (a survey of alternative set theories);
- Cellular automata;
- Classical nonclassical logics and standard nonstandard models;
- Fortune-teller is of no use (what can and what cannot be predicted),
- Conditional probability and other probabilistic notions;
- Introduction to game theory;
- Sequences, close to periodical (with Yu.Pritykin).

## Languages spoken

- Russian (native)
- English (advanced)
- French (basic)